Procedia Computer Science
Volume 51, 2015, Pages 2703–2707
ICCS 2015 International Conference On Computational Science

Comparison of Two Diversification Methods to Solve the Quadratic Assignment Problem

Omar Abdelkafi, Lhassane Idoumghar, and Julien Lepagnot

University of Haute-Alsace, Mulhouse, Alsace, France
omar.abdelkafi@uha.fr,lhassane.idoumghar@uha.fr,julien.lepagnot@uha.fr

Abstract
The quadratic assignment problem is one of the most studied NP-hard problems. It is known for its complexity which makes it a good candidate for the parallel design. In this paper, we propose and analyze two parallel cooperative algorithms based on hybrid iterative tabu search. The only difference between the two approaches is the diversification methods. Through 15 of the hardest well-known instances from QAPLIB benchmark, our algorithms produce competitive results.

Keywords: metaheuristics, hybrid iterative tabu search, diversification algorithms

1 Introduction

The Quadratic Assignment Problem (QAP) is an NP-hard problem. It is well known for its multiple applications in various fields. The QAP was first introduced by Koopmans and Beckmann [5] to model a facility location problem. The objective is to find a minimum cost assignment of facilities to locations considering the flow of materials between facilities and the distance between locations.

In this work, we propose and analyze two parallel Hybrid Iterative Tabu Search (HITS) algorithms using cooperation strategies. Our main objective is to compare the efficiency of information exchange with well-known diversification methods. The rest of the paper is organized as follows. In section 2, we review some of the best-known approaches to solve the QAP. In section 3, we describe the parallel design of our algorithms. Section 4 shows the experimental results. Finally, in section 5, we conclude the paper and we propose some perspectives.

2 Background and Related Works

Since its introduction in 1957 [5], the QAP became an important problem in theory and practice. It can be considered as one of the hardest combinatorial problems due to its computational complexity.

Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2015

doi:10.1016/j.procs.2015.05.392

One of the first efficient approach to solve the QAP is the Robust Tabu search (Ro-Ts) proposed by Taillard in 1991 [7]. In 2013, the work of U. Benlic et al [1] proposes an Iterative Local Search (ILS) with a breakout strategy based on the history of the search (BLS). Recently, [2] presented a memetic algorithm (BMA) for the well-known QAP. They combines the previous BLS [1] with the standard uniform crossover.

E.G. Talbi [8] classified the parallel design of metaheuristics into 3 levels (algorithmic level, iteration level and solution level). In this work, we use only the algorithmic level.

3 Distance Cooperation Hybrid Iterative Tabu Search

Two versions are proposed in this work: The DIStance COoperation between Hybrid Iterative Tabu Search with the Glover Diversification (*DISCO-HITS-GD*) and The DIStance COoperation between Hybrid Iterative Tabu Search with the Uniform crossover diversification (*DISCO-HITS-UX*).

Different Hybrid Iterative Tabu Search (HITS) are executed in parallel from different starting solutions. In our proposition, there is an exchange of information between a set of processes in parallel following a ring topology. Each process executes one HITS and uses the history of the search of its neighbor process. It means that the evolution of each HITS depends on the search history of its neighbor HITS. The aim is to explore different regions of the space. The process sends its current solution and receives the current solution of the neighbor process. According to the distance between these two solutions, each HITS takes a decision and follows a specific series of instructions to continue the search.

For each process, a succession of TS [7] are executed. Algorithm 1 computes the difference between the two neighbor solutions to determine the distance between them. According to the distance, the algorithm takes one decision, to execute a diversification (glover diversification for DISCO-HITS-GD or uniform crossover for DISCO-HITS-UX), to perturb the solution (line 29 algorithm 1) or to make a re-localization of this solution (line 31 algorithm 1).

For the DISCO-HITS-GD, our approach applies the diversification proposed by Glover [3] called the Glover Diversification (GD) in this work. The diversification procedure takes a solution and executes a set of permutations following a step value.

For the DISCO-HITS-UX, the diversification applied is the uniform crossover (UX). The diversification procedure takes a solution and executes the UX following the sequence given by a random vector.

4 Experimental Results

4.1 Platform and Tests

In our experimentation, the algorithms are written in C/C++ and run on a cluster of 10 machines Intel Core processor i5-3330 CPU (3.00GHz) with 4 GB of RAM. The proposed algorithms are experimented on benchmark instances from the QAPLIB (*http://www.seas.upenn.edu/qaplib/inst.html*). All the results are expressed as a percentage deviation from the Best Known Solutions (BKS) (eq 1). All the BKS can be found in the online benchmark library QAPLIB. Each instance is executed 10 times and the average results of these executions are given.

$$deviation = \frac{(solution - BKS) \times 100}{BKS} \tag{1}$$

Algorithm 1 Distance cooperation between hybrid iterative tabu search

```
Input: perturb: % perturbation; n: size of solution; cost: cost of the current solution;
    Fcost: best cost found; solution: current solution; Fsolution: best solution found;
    Information: solution exchanged;
Initialization of the solution for the current process;
repeat
    TS algorithm
    if cost < Fcost then
        Fcost = cost; Update the Fsolution with solution;
    end if
    level = 0; counter = 0;
    Update the Information with solution;
    Exchange Information between process (ring topology);
    for i = 0 to n /* Compute distances */ do
        if solution[i] == Information[i] then
            counter ++;
        end if
    end for
    if counter < n/4 then
        level = 0; /* Big distance between the two processes */
    else
        if counter < (3×n)/4 then
            level = 1; /* Processes are relatively close */
        else
            level = 2; /* Processes are very close */
        end if
    end if
    if level == 0 then
        Update solution with a Diversification of Fsolution;
    else
        if level == 1 then
            Perturbation of solution with the perturb parameter;
        else
            Re-localization of solution;
        end if
    end if
until (Stop condition)
```

The QAPLIB archive comprises 134 instances that can be classified into four types.

4.2 Comparison of the Diversification Methods

We compare the variants DISCO-HITS-GD and DISCO-HITS-UX to see precisely which diversification techniques converge better. The value obtained by each algorithm is recorded every 20 global iterations. The UX method is more efficient than the GD method to explore the search space of the QAP for our algorithm design. The DISCO-HITS-UX algorithm has the capacity to get better results.

Table 1: Comparison of DISCO-HITS-GD and DISCO-HITS-UX with BMA, BLS, CPTS and PILS

Instances(15)	**BKS**	**DISCO-HITS-GD**		**DISCO-HITS-UX**		**BMA**		**BLS**		**CPTS**		**PILS**	
		deviation	times	deviation	times	deviation	times	deviation	times	deviation	times	deviation	times
sko100a	152002	**0.000(10)**	53.65	**0.000(10)**	52.41	0.000(10)	22.3	0.001(9)	20.8	0.000(10)	304.8	0.012(3)	7.9
sko100b	153890	**0.000(10)**	51.50	**0.000(10)**	52.63	0.000(10)	6.5	0.000(10)	10.8	0.000(10)	309.6	0.007(5)	7.3
sko100c	147862	**0.000(10)**	51.52	**0.000(10)**	52.04	0.000(10)	12.0	0.000(10)	15.5	0.000(10)	316.1	0.002(6)	11.5
sko100d	149576	**0.000(10)**	54.09	0.000(8)	51.28	0.006(9)	20.9	0.001(5)	38.9	0.000(10)	309.8.6	0.021(0)	11.8
sko100e	149150	0.001(9)	51.42	0.001(9)	51.30	**0.000(10)**	11.9	**0.000(10)**	42.5	**0.000(10)**	309.1	0.001(7)	6.8
sko100f	149036	0.001(8)	51.48	0.001(9)	51.64	**0.000(10)**	23.0	**0.000(10)**	17.3	0.003(4)	310.3	0.037(0)	11.7
wil100	273038	0.000(9)	51.60	0.000(8)	51.62	**0.000(10)**	14.5	**0.000(10)**	18.9	**0.000(10)**	316.6	0.004(1)	6.3
tho150	8133398	0.010(0)	218.95	**0.004(0)**	199.36	0.008(3)	416.4	0.023(1)	268.8	0.013(0)	1991.7	0.068(0)	36.2
tai40a	3139370	0.030(6)	3.18	**0.007(9)**	3.22	0.059(2)	8.1	0.022(7)	38.9	0.148(1)	3.5	0.280(0)	12.0
tai50a	4938796	0.062(8)	6.19	**0.048(8)**	6.29	0.131(2)	42.0	0.157(2)	45.1	0.440(0)	10.3	0.663(0)	11.2
tai60a	7205962	0.303(0)	10.71	0.272(0)	10.77	**0.144(2)**	67.5	0.251(1)	47.9	0.476(0)	26.4	0.820(0)	7.4
tai80a	13515450	0.573(0)	25.54	0.561(0)	25.62	**0.426(0)**	65.8	0.517(0)	47.3	0.691(0)	94.8	0.927(0)	12.7
tai100a	21052466	0.552(0)	52.07	**0.359(0)**	52.36	0.405(0)	44.1	0.430(0)	39.0	0.589(0)	261.2	1.027(0)	9.8
tai100b	1185996137	0.000(9)	51.26	**0.000(10)**	53.08	0.000(10)	13.6	0.000(10)	16.0	0.001(8)	241.0	0.000(10)	2.3
tai150b	498896643	**0.015(0)**	214.26	0.027(3)	214.86	0.060(1)	78.1	0.075(1)	243.6	0.076(0)	7377.8	0.095(0)	36.7
Average		0.108(89)	56.91	0.085(94)	61.90	**0.083(89)**	70.39	0.098(83)	60.75	0.162(73)	812.20	0.264(32)	12.77
Average type 4		0.002(66)	61.93	**0.001(64)**	70.29	0.002(72)	87.37	0.003(65)	54.19	0.002(64)	521	0.019(22)	12.44
Average type 2		0.373(14)	19.64	0.249(17)	19.65	**0.233(6)**	45.5	0.275(10)	43.64	0.469(1)	79.24	0.743(0)	10.62
Average type 3		**0.008(9)**	130.03	0.014(13)	133.97	0.030(11)	64.71	0.038(8)	129.8	0.039(8)	3809.4	0.048(10)	19.5

4.3 Literature Comparison

Table 1 presents several comparisons of DISCO-HITS-GD and DISCO-HITS-UX with four leading algorithms from the literature: BMA[2]; BLS[1]; CPTS[4]; PILS[6]. The focus of this comparison is the quality of solutions. We use 15 well-known benchmark instances from the QAPLIB which are difficult to solve.

For the experiments of table 1, the best global average is obtained by the BMA [2] algorithm with 0.083%. It is followed by DISCO-HITS-UX with very close average results of 0.085%. DISCO-HITS-UX outperforms the average of the other three algorithms BLS [1], CPTS [4] and PILS [6]. The QAPLIB benchmark instances are classified into 4 types. For the instances of type 1, which are the real life instances, all our algorithms can solve them easily. The unstructured randomly generated instances based on a uniform distribution (type 2) are the hardest to solve. BMA is very efficient for this particular class (type 2), however, DISCO-HITS-GD can outperform BMA on two of these instances and DISCO-HITS-UX outperforms BMA on three instances among the 5. Our best variant DISCO-HITS-UX gets an average of 0.249% on these 5 instances and it outperforms [1], [4] and [6]. For the type 3, DISCO-HITS-UX and DISCO-HITS-GD outperform all the 4 algorithms of the literature presented on table 1. Finally, for the type 4, DISCO-HITS-UX gets better results than the 4 algorithms of table 1.

5 Conclusion and Perspectives

In this work, we have presented and validated two variants of a parallel HITS to solve the QAP. We have evaluated our approaches on 15 benchmark instances from the QAPLIB. It demonstrate high-quality results on the set of well-known benchmark instances from QAPLIB. The comparison between the variants has revealed the potential of the exchange of information in the parallel design and it has shown the efficiency of the UX to solve the QAP.

As a future work, there are several possible ways to extend this work. One possibility is to experiment other parameters to get better results. Another possibility is to explore the two other parallel designs (the iteration level and the solution level). The GPU platform is appropriate for the implementation of these two levels thanks to its single instruction multiple data architecture.

References

[1] U. Benlic and J.K. Hao. Breakout local search for the quadratic assignement problem. *Applied Mathematics and Computation*, 219(9):4800–4815, 2013.

[2] U. Benlic and J.K. Hao. Memetic search for the quadratic assignment problem. *Expert Systems with Applications*, 42:584–595, 2015.

[3] F. Glover. A template for scatter search and path relinking. *Lecture Notes in Computer Science*, 1363:1–51, 1998.

[4] T. James, C. Rego, and F. Glover. A cooperative parallel tabu search algorithm for the quadratic assignment problem. *European Journal of Operational Research*, 195(3):810–826, 2009.

[5] T. Koopmans and M. Beckmann. Assignment problems and the location of economic activities. *Econometrica*, 25(1):53–76, 1957.

[6] T. Stützle. Iterated local search for the quadratic assignment problem. *European Journal of Operational Research*, 174(3):1519–1539, 2006.

[7] E. Taillard. Robust taboo search for the quadratic assignement problem. *Parallel computing*, 17(4-5):443–455, 1991.

[8] El-Ghazali Talbi. *Metaheuristics: from Design to Implementation.* John wiley and sons Inc, 2009.